NAME

getopt − get option letter from argv

SYNOPSIS

int getopt (argc, argv, optstring)
int argc;
char **argv;
char *optstring;
extern char *optarg;
extern int optind;

DESCRIPTION

Getopt returns the next option letter in *argv* that matches a letter in *optstring*. *Optstring* is a string of recognized option letters; if a letter is followed by a colon, the option is expected to have an argument which may or may not be separated from it by white space. *Optarg* is set to point to the start of the option argument on return from *getopt*.

Getopt places in *optind* the *argv* index of the next argument to be processed. Since *optind* is external, it is normally initialized to zero automatically before the first call to *getopt*.

When all options have been processed (i.e., up to the first non-option argument), *getopt* returns **EOF**. The special option "−−" may be used to delimit the end of the options; **EOF** will be returned, and "−−" will be skipped.

DIAGNOSTICS

Getopt prints an error message on *stderr* and returns a question mark ('?') when it encounters an option letter not included in *optstring*.

EXAMPLE

The following code fragment shows how one might process the arguments for a command that can take the mutually exclusive options **a** and **b**, and the options **f** and **o**, which require arguments.

```
main (argc, argv)
        int argc;
        char **argv;
{
        int c;
        extern int optind;
        extern char *optarg;
        .
        .
        .
        while ((c = getopt (argc, argv, "abf:o:")) != EOF)
                switch (c) {

                case 'a':
                        if (bflg)
                                errflg++;
                        else
                                aflg++;
                        break;

                case 'b':
                        if (aflg)
                                errflg++;
                        else
                                bproc();
                                break;

                        case ´f´:
                                ifile = optarg;
                                break;

                        case ´o´:
                                ofile = optarg;
                                bufsiza = 512;
                                break;

                        case ´?´:
                                errflg++;
                        }
                if (errflg) {
                        fprintf (stderr, "usage: . . . ");
                        exit (2);
                }
                for( ; optind < argc; optind++) {
                        if (access (argv[optind], 4)) {
                .
                .
                .
        }
```